10. A method for creating a data base representing a virtual world, the method comprising:

receiving a plurality of polygon representations of virtual objects;

selecting first and second virtual objects from said plurality of polygon representations of virtual objects;

grouping the first and second virtual objects into a three-dimensional grouped object;

assigning a grouping hierarchy for the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and

calculating an orientation and position of the child object relative to the first virtual object.

11. The method of claim 10, wherein said grouping includes representing the grouped object by at least one of the following:

a three-dimensional and rotatable wireframe object, and

a three-dimensional and rotatable polygon object.

12. The method as recited in claim 10, further comprising:

assigning an origin on the first virtual object around which the second virtual object can rotate; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the first virtual object.

13. The method as recited in claim 10, further comprising:

assigning an origin to the first virtual object, wherein said origin has a fixed position relative to said first virtual object; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the origin.

14. The method as recited in claim 13, wherein the origin is within the first object.

15. The method as recited in claim 13, wherein the origin is on the surface of the first object.

16. The method as recited in claim 13, wherein the origin is a predetermined distance from the surface of the first object.

17. The method as recited in claim 13, wherein the second virtual object has one degree of freedom relative to said origin.

18. The method as recited in claim 13, wherein the second virtual object has at least two degrees of freedom relative to said origin.

19. The method as recited in claim 10, further comprising:

assigning an axis to the first virtual object, wherein said axis has a fixed position relative to said first virtual object; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the axis.

20. The method as recited in claim 19, wherein the axis is a line.

21. The method as recited in claim 19, wherein the axis is a line segment.

22. The method as recited in claim 19, wherein the axis is at least partially within the first object.

23. The method as recited in claim 19, wherein the axis is at least partially on the surface of the first object.

24. The method as recited in claim 19, wherein the axis does not intersect the first object.

25. The method as recited in claim 13, wherein the second virtual object has one degree of freedom relative to said axis.

26. The method as recited in claim 13, wherein the second virtual object has at least two degrees of freedom relative to said axis.

27. The method as recited in claim 10, further comprising:

defining a locus of points having a fixed position relative to said first virtual object and said origin; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the locus of points.

28. The method as recited in claim 12, further comprising specifying a minimum angle and a maximum angle that the second virtual object can rotate with respect to the origin.

29. The method as recited in claim 12, further comprising specifying one or more angles as constraints on the rotation of the second object relative to said origin.

30. The method as recited in claim 12, further comprising specifying one or more constraints on the translation of the second object relative to said origin.

31. The method as recited in claim 10, further comprising:

receiving a third virtual object;

assigning an origin on the first virtual object around which the third virtual object can rotate; and

assigning a three-dimensional constraint of motion to the third virtual object that constrains how the third virtual object can rotate with respect to the first virtual object.

32. The method as recited in claim 10, further comprising:

receiving a third virtual object; and

grouping said third object into the three-dimensional grouped object.

33. The method as recited in claim 32, wherein the third virtual object is assigned as the child of the first virtual object; and wherein the orientation and position of the third virtual object is calculated relative to the first virtual object.

34. The method as recited in claim 32, wherein the third virtual object is assigned as the child of the second virtual object; and wherein the third virtual object inherits the constraints assigned to the second virtual object.

35. The method as recited in claim 34, further comprising calculating the third virtual object's orientation and position relative to the first virtual object.

36. The method as recited in claim 35, further comprising calculating the third virtual object's orientation and position relative to the second virtual object.

37. The method as recited in claim 31, further comprising specifying a minimum angle and a maximum angle that the third virtual object can rotate with respect to the origin.

38. The method as recited in claim 10, further comprising:

grouping a fourth virtual object and a fifth virtual object to create a second grouped object; and

grouping the second grouped object with first grouped object; and

assigning a grouping hierarchy for the fourth and fifth virtual objects, wherein the fourth and fifth virtual objects are assigned as the child of the second virtual object.

39. The method as recited in claim 10, further comprising assigning color values to the grouped object, wherein each virtual object that is part of said grouped object inherits the assigned color.

40. The method as recited in claim 10, further comprising assigning a color value to a particular virtual object within the grouped object, wherein each virtual object that is a child of the particular virtual object inherits the assigned color.

41. The method as recited in claim 10, further comprising assigning texture values to the grouped object.

42. The method as recited in claim 10, further comprising coupling real world data to the grouped object.

43. The method as recited in claim 10, wherein the real world data comprises positional and rotational information.

44. The method as recited in claim 10, wherein said grouping the first and second virtual objects includes:

selecting a first edge of said first virtual object;

selecting a second edge of said second virtual object;

wherein said three-dimensional grouped object comprises said first and second virtual objects joined with at least a portion of said first edge of said first virtual object contacting at least a portion of said second edge of said second virtual object.

45. The method as recited in claim 10, wherein said grouping the first and second virtual objects includes:

selecting a first edge of said first virtual object;

selecting a second edge of said second virtual object;

wherein said three-dimensional grouped object comprises said first and second virtual objects joined at an intersection of the first and second edges.

46. ~~A memory media~~ comprising program instructions for creating ~~a data base~~ representing a virtual world, wherein the program instructions are executable to implement:

receiving a plurality of polygon representations of virtual objects;

selecting first and second virtual objects from said plurality of polygon representations of virtual objects;

grouping the first and second virtual objects into a three-dimensional grouped object;

assigning a grouping hierarchy for the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and

~~calculating an orientation and position of the child object relative to the first virtual object.~~

47. ~~The memory medium of claim 46, wherein said grouping includes representing the grouped object by at least one of the following:~~

~~a three-dimensional and rotatable wireframe object, and~~

~~a three-dimensional and rotatable polygon object.~~

48. The memory medium as recited in claim 46, wherein the program instructions are further executable to implement:

assigning an origin to the first virtual object; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can translate and rotate with respect to the first virtual object.

49. The memory medium as recited in claim 48, wherein the program instructions are further executable to implement:

specifying one or more angles that constrain the second virtual object's rotation with respect to the origin.

50. The memory medium as recited in claim 48, wherein the program instructions are further executable to implement:

specifying one or more angles relative to the origin that constrain the second virtual object's freedom to move relative to the origin.

51. The memory medium as recited in claim 48, wherein the program instructions are further executable to implement:

receiving a third virtual object; and

grouping the third object into the grouped object as a child of the second object, wherein the third object inherits the second object's constraints relative the to origin.

52. The memory medium as recited in claim 51, wherein the program instructions are further executable to implement:

receiving a third virtual object;

grouping the third object into the grouped object; and

specifying constraint angles that the third virtual object can rotate with respect to the origin.

53. The memory medium as recited in claim 46, wherein the program instructions are further executable to implement assigning color values to the grouped object, wherein each virtual object in the grouped object inherits the color values

54. The memory medium as recited in claim 46, wherein the program instructions are further executable to implement assigning texture values to the grouped object, wherein each virtual object in the grouped object inherits the texture values

55. The memory medium as recited in claim 46, wherein the program instructions are further executable to implement coupling real world data to the grouped object.

56. The memory medium as recited in claim 46, wherein said grouping the first and second virtual objects includes:

selecting a first edge of said first virtual object;

selecting a second edge of said second virtual object;

wherein said three-dimensional grouped object comprises said first and second virtual objects joined with at least a portion of said first edge of said first virtual object contacting at least a portion of said second edge of said second virtual object.

57. The memory medium as recited in claim 46, wherein said grouping the first and second virtual objects includes:

selecting a first edge of said first virtual object;

selecting a second edge of said second virtual object;

wherein said three-dimensional grouped object comprises said first and second virtual objects joined at an intersection of the first and second edges.

58. A system for creating a data base representing a virtual world, the system comprising:

a computer system comprising a CPU and memory, wherein the memory stores a plurality of polygon representations of virtual objects;

a user input device coupled to the computer system for providing user input to the computer system;

wherein the CPU is operable to select first and second virtual objects from said plurality of polygon representations of virtual objects;

wherein the CPU is operable to group the first and second virtual objects into a three-dimensional grouped object;

wherein the CPU is operable to assign a grouping hierarchy for the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and

wherein the CPU is operable to calculate an orientation and position of the child object relative to the first virtual object.

59. The system as recited in claim 58, wherein the CPU is operable to represent the grouped object as at least one of the following:

a three-dimensional and rotatable wireframe object, and

a three-dimensional and rotatable polygon object.

60. The system as recited in claim 58, wherein the CPU is operable to:

assign an origin on the first virtual object around which the second virtual object can rotate, and

assign a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the first virtual object.

61. The system as recited in claim 60, wherein the CPU is configured to receive constraint data from the user input device specifying a minimum angle and a maximum angle that the second virtual object can rotate with respect to the origin, wherein the CPU is configured to constrain the motion of the second virtual object with respect to the origin in response to receiving said constraint data.

62. The system as recited in claim 58, wherein the CPU is operable to:

assign an origin on the first virtual object around which a third virtual object can rotate, and

assign a three-dimensional constraint of motion to the third virtual object that constrains how the second virtual object can rotate with respect to the first virtual object.

63. The system as recited in claim 62, wherein the CPU is configured to receive constraint data from the user input device specifying a minimum angle and a maximum angle that the third virtual

object can rotate with respect to the origin, wherein the CPU is configured to constrain the motion of the third virtual object with respect to the origin in response to receiving said constraint data.

64. The system as recited in claim 58, wherein the CPU is further configured to assign color values to the grouped object.

65. The system as recited in claim 58, wherein the CPU is further configured to assign texture values to the grouped object.

66. The system as recited in claim 58, wherein the CPU is further configured to couple real world data to the grouped object.

67. The system as recited in claim 58, wherein the CPU is further configured to:

select a first edge of said first virtual object, and

select a second edge of said second virtual,

wherein said three-dimensional grouped object comprises said first and second virtual objects joined with at least a portion of said first edge of said first virtual object contacting at least a portion of said second edge of said second virtual object.

68. The system as recited in claim 58, wherein said CPU is configured to group the first and second virtual objects by:

selecting a first edge of said first virtual object, and

selecting a second edge of said second virtual object,

wherein said three-dimensional grouped object comprises said first and second virtual objects joined at an intersection of the first and second edges.

69. ~~A method for creating a data base representing a virtual world, the method comprising:~~

receiving a plurality of polygon representations of virtual objects;

selecting first and second virtual objects from said plurality of polygon representations of virtual ~~objects;~~

grouping the first and second virtual objects into a grouped object comprising a combination of the first and second virtual objects, wherein the first and second virtual objects intersect; and

representing the grouped object by at least one of the following:

a three-dimensional and rotatable wireframe object, and

~~a three-dimensional and rotatable polygon object.~~

70. The method as recited in claim 69, further comprising:

assigning a grouping hierarchy for the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and

calculating an orientation and position of the child object relative to the first virtual object.

71. The method as recited in claim 70, further comprising:

assigning group attributes to the grouped object; and

assigning individual attributes to a particular virtual object within the grouped object, wherein the individual attributes are inherited by child objects of the particular virtual object.

72. The method as recited in claim 70, further comprising:

assigning an origin on the first virtual object around which the second virtual object can rotate; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the first virtual object.

73. The method as recited in claim 72, further comprising specifying one or more angles that constrain the second virtual object's rotation with respect to the origin.

74. The method as recited in claim 72, further comprising specifying one or more constraint values that constrain the second virtual object's translation with respect to the origin.

75. The method as recited in claim 74, further comprising:

grouping a third virtual object into the grouped object; and

assigning a three-dimensional constraint of motion to the third virtual object that constrains how the third virtual object can rotate with respect to the origin.

76. The method as recited in claim 75, further comprising specifying one or more constraint angles that constrain the third virtual object's rotation with respect to the origin.

77. The method as recited in claim 70, further comprising assigning texture and color values to the grouped object.

78. The method as recited in claim 70, further comprising coupling real world data to the grouped object.

79. The method as recited in claim 78, wherein said coupling comprises:
taking sensor data from real world inputs; and
varying the orientation of the child object with respect to the origin in relation to the sensor data.

80. The method as recited in claim 78, wherein said coupling comprises:
taking sensor data from real world inputs; and
varying the position of the child object with respect to the origin in relation to the sensor data.

81. The method as recited in claim 78, wherein said coupling comprises:
sensing the relative position of two real world physical objects; and
adjusting the relative position of the second virtual object relative to the origin accordingly.

82. The method as recited in claim 78, wherein said coupling comprises:
sensing the relative orientation of two real world physical objects; and
adjusting the relative orientation of the second virtual object relative to the origin accordingly.

83. The method as recited in claim 82, wherein said sensing is accomplished using a data glove.

Sub C3

84. ~~A computer program for creating~~ a virtual world ~~data base, wherein said computer~~ program is embodied on computer-readable media and comprises instructions configured to:
store polygon representations of a plurality of virtual objects;
select a first virtual object and a second virtual object from said plurality of virtual objects;
assign attributes to the first and second virtual objects;
group said first and second virtual objects into a grouped object, wherein said first and second virtual objects intersect;
represent the grouped object by at least one of the following:
a three-dimensional and rotatable wireframe object, and
a three-dimensional and rotatable polygon object;
assign a grouping hierarchy to the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and
~~calculate an orientation and position of the child object relative to the first virtual object.~~

85. The computer program as recited in claim 84, wherein said computer program further comprises instructions configured to:
assign an origin on the first virtual object around which the second virtual object can rotate; and

assign a three-dimensional constraint of motion to the second virtual object, wherein said three-dimensional constraint of motion constrains how the second virtual object can rotate with respect to the first virtual object.

86. The computer program as recited in claim 84, wherein said computer program further comprises instructions configured to specify a minimum angle and a maximum angle that the third virtual object can rotate with respect to the origin.

87. The computer program as recited in claim 84, wherein said computer program further comprises instructions configured to assign color values to the grouped object.

88. The computer program as recited in claim 84, wherein said computer program further comprises instructions configured to assign texture values to the grouped object.

89. The computer program as recited in claim 84, wherein said computer program further comprises instructions configured to couple real world data to the grouped object.

90. The computer program as recited in claim 89, wherein said computer program is configured to:

take sensor data from real world inputs; and

vary the orientation of the child object with respect to the origin in relation to the sensor data.

91. The computer program as recited in claim 89, wherein said computer program is configured to:

take sensor data from real world inputs; and

vary the position of the child object with respect to the origin in relation to the sensor data.

92. The computer program as recited in claim 89, wherein said computer program is configured to:

sense the relative position of two real world physical objects; and

adjust the relative position of the second virtual object relative to the origin accordingly.

93. The computer program as recited in claim 89, wherein said computer program is configured to:

sense the relative orientation of two real world physical objects; and

adjust the relative orientation of the second virtual object relative to the origin accordingly.

94. The computer program as recited in claim 93, wherein said computer program is configured to sense the relative orientation of two real world objects from input signals received from a data glove.

95. ~~An apparatus for creating a virtual world data base, comprising:~~

a receiving means for receiving first and second polygon representations of respective first and second virtual objects in a virtual world;

a selecting means coupled to said receiving means and configured to select said first and second virtual objects;

a grouping means coupled to said receiving means and selecting means, wherein said grouping means is configured to group said first and second virtual objects into a grouped object, wherein the grouped object is represented by at least one of a three-dimensional and rotatable wireframe object and a three dimensional and rotatable polygon object; and

an attribute assigning means coupled to said grouping means, wherein said assigning means is configured to assign an attribute to the first and second virtual objects, wherein the attribute assigning means comprises a hierarchy means for assigning a grouping hierarchy to the first and second virtual objects, wherein the second virtual object is assigned as a child object of the first virtual object, and wherein an orientation and a position of the child object is calculated relative to the first virtual object.

96. The apparatus as recited in claim 95, wherein said selecting means selects said first and second virtual objects by selecting one edge from each of said first and second virtual objects.

97. The apparatus as recited in claim 95, wherein said attribute assigning means further comprises:

an origin assigning means for assigning an origin on the first virtual object around which a third virtual object can rotate, wherein said third virtual object is selected by said selecting means from said plurality of virtual objects; and

a constraint assigning means for assigning a three-dimensional constraint of motion to the third virtual object to constrain how the third virtual object can rotate with respect to the first virtual object.

98. The apparatus as recited in claim 97, wherein the constraint assigning means is further configured to specify a minimum angle and a maximum angle that said third virtual object can rotate with respect to said origin.

99. The apparatus as recited in claim 97, further comprising an color assigning means coupled to the attribute assigning means, wherein said color assigning means is configured to assign color values to the grouped objects.

100. The apparatus as recited in claim 97, further comprising a texture assigning means coupled to the attribute assigning means, wherein the texture assigning means is configured to assign texture values to the grouped objects.

101. The apparatus as recited in claim 97, further comprising a data coupling means coupled to the attribute assigning means, wherein the data coupling means is configured to couple real world data to the grouped objects.

102. A method for creating a data base representing ~~a virtual world, the method comprising:~~

receiving a plurality of polygon representations of virtual objects;

selecting first and second virtual objects from said plurality of polygon representations of virtual objects;

grouping the first and second virtual objects into a hierarchical grouped object, wherein said grouping includes:

selecting a first mathematical edge of said first virtual object;

selecting a second mathematical edge of said second virtual object; and

representing the grouped object by at least one of the following:

a three-dimensional and rotatable hierarchical wireframe object, and

a three-dimensional and rotatable hierarchical polygon object.

103. The method of claim 102, wherein said first and second mathematical edges are single points.

104. The method of claim 102, wherein said first and second mathematical edges are detached from said first and second virtual objects.

105. The method of claim 102, wherein the first and second virtual objects intersect, and wherein the grouped object comprises said first and second virtual objects joined with at least a portion of said first edge of said first virtual object contacting at least a portion of said second edge of said second virtual object.

106. ~~The~~ method of claim 102, ~~further comprising:~~

assigning a grouping hierarchy for the first and second virtual objects, wherein the second virtual object is assigned as the child of the first virtual object; and

calculating an orientation and position of the child object relative to the first virtual object.

assigning an origin on the first virtual object around which the second virtual object can rotate; and

assigning a three-dimensional constraint of motion to the second virtual object that constrains how the second virtual object can rotate with respect to the first virtual object.

107. The method as recited in claim 102, further comprising specifying a minimum angle and a maximum angle that the second virtual object can rotate with respect to the origin.

108. The method as recited in claim 102, further comprising assigning an attributes to the grouped object, wherein the attribute is texture, color, normal direction, maximum rotation angle, or minimum rotation angle.